AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listing of claims in the above-referenced application.

Listing of Claims:

1.   (Original) A method for synchronizing a plurality of endpoints in a data storage system at a first synchronization point, the method comprising:

performing, by a first endpoint, a synchronization start operation wherein a first message is sent from said first endpoint to one or more other endpoints in the data storage system, said first message including a first key value corresponding to said first synchronization point representing a current processing state of said first endpoint;

determining, by said first endpoint, a timeout period;

determining, by said first endpoint using processing state information as reported to said first endpoint by other endpoints, whether synchronization with a selected portion of said one or more other endpoints at said first synchronization point has been accomplished within said timeout period; and

if said first endpoint determines that synchronization has not been accomplished within said timeout period, sending a second message to said one or more other endpoints indicating that said first endpoint is at another synchronization point different from said first synchronization point.

2.   (Currently Amended) The method of Claim 1, wherein said plurality of endpoints are directors in said data storage system, said synchronization start operation defines a start of a synchronization period for the first endpoint, and said second message is sent as part of a

synchronization stop operation marking an end of the synchronization period for the first endpoint.

3.   (Original) The method of Claim 2, wherein said first and second messages are synchronization messages sent using a message switch included in said data storage system.

4.   (Original) The method of Claim 3, further comprising:

storing, by said first endpoint, said processing state information in a key value storage area local to said first endpoint.

5.   (Original) The method of Claim 4, further comprising:

broadcasting, by each endpoint in the data storage system, a synchronization message including a key value corresponding to a current processing state of said each endpoint to every other endpoint in the data storage system; and

recording, by each endpoint in the data storage system, key values for each endpoint for all received synchronization messages, each endpoint storing said key values in a key value storage area local to each endpoint.

6.   (Original) The method of Claim 5, wherein said determining whether synchronization with a selected portion of said one or more other endpoints at said first synchronization point has been accomplished within said timeout period is performed as part of a synchronization check operation.

7. (Original) The method of Claim 6, wherein said second message is a synchronization message including a key value corresponding to said other synchronization point.

8. (Original) The method of Claim 7, wherein said key value indicates one of: an invalid synchronization processing state, and another valid processing state of said first endpoint.

9. (Original) The method of Claim 1, wherein said first endpoint determines whether synchronization has occurred for a subset of said selection portion of endpoints in accordance with said processing state information of said first endpoint.

10. (Original) The method of Claim 1, wherein said selected portion of said one or more other endpoints is stored by said first endpoint as a synchronization mask.

11. (Original) The method of Claim 10, wherein said synchronization mask includes a number of bits equal to a maximum number of possible endpoints in a data storage system.

12. (Original) The method of Claim 1, wherein said timeout period is determined in accordance with an operation being performed for which said first synchronization point is specified.

13. (Original) The method of Claim 1, wherein each endpoint in said data storage system maintains a local copy of state information representing a processing state of each endpoint as reported by each endpoint in synchronization messages sent over a message switch.

14. (Original) The method of Claim 13, wherein each local copy of state information as maintained by each endpoint is not synchronized with other local copies of state information as maintained by other endpoints.

15. (Currently Amended) ~~The method of Claim 14,~~

A method for synchronizing a plurality of endpoints in a data storage system at a first synchronization point, the method comprising:

performing, by a first endpoint, a synchronization start operation wherein a first message is sent from said first endpoint to one or more other endpoints in the data storage system, said first message including a first key value corresponding to said first synchronization point representing a current processing state of said first endpoint;

determining, by said first endpoint, a timeout period;

determining, by said first endpoint using processing state information as reported to said first endpoint by other endpoints, whether synchronization with a selected portion of said one or more other endpoints at said first synchronization point has been accomplished within said timeout period; and

if said first endpoint determines that synchronization has not been accomplished within said timeout period, sending a second message to said one or more other endpoints indicating that said first endpoint is at another synchronization point different from said first synchronization point, and wherein each endpoint in said data storage system maintains a local copy of state information representing a processing state of each endpoint as reported by each endpoint in synchronization messages sent over a message switch, and each local copy of state information as maintained by each endpoint is not synchronized with other local copies of state information as maintained by other endpoints, and

wherein said first endpoint is any one of said plurality of endpoints which reaches said first synchronization point prior to other endpoints included in said selection portion in accordance with said processing state information, said processing state information being a private copy of synchronization information associated with said first endpoint.

16. (Original) The method of Claim 15, wherein each of said endpoints in said selected portion perform synchronization in accordance with a peer to peer model.

17. (Original) The method of Claim 1, wherein said first synchronization point represents an event corresponding to one of: enabling and disabling a mirrored write operation wherein when enabled, a write to a portion of a global memory is performed to two memory boards, and when disabled, a write to a portion of a global memory is performed to one memory board.

18. (Original) The method of Claim 1, wherein said first synchronization point represents an event corresponding to one of: enabling and disabling operations to a hardware element.

19. (Original) The method of Claim 18, further comprising:
disabling operations to said hardware element when said hardware element is brought off-line.

20. (Original) The method of Claim 19, further comprising:

enabling operations to said hardware element when said hardware element is brought on-line.

21.   (Original) A method executed in a data storage system for synchronizing execution at a first synchronization point between a plurality of processors in said data storage system, the method comprising:

performing a synchronization start operation by a first processor defining a beginning of a synchronization period for said first processor, said synchronization start operation including said first processor performing:

sending a first synchronization message to all other processors in said data storage system, said first synchronization message include a first key value corresponding to said first synchronization point representing a current processing state of said first processor;

determining a portion of said plurality of processors with whom said first processor is to synchronize at said first synchronization point;

determining whether synchronization at said first synchronization point has been accomplished for said portion of processors; and

determining an expiration period indicating a maximum amount of time for which said first processor will wait for synchronization at said first synchronization point for said portion of processors;

receiving one or more other synchronization messages from other ones of said plurality of processors in said data storage system wherein each of said one or more other synchronization messages includes a key value indicating a processing state of its sending processor;

storing, by said first processor in a key value storage local to said first processor, key values from said one or more other synchronization messages, wherein said first processor uses key values in said key value storage to determine whether synchronization at said first synchronization point has been accomplished for said portion of processors as part of a synchronization check operation; and

performing by said first processor a synchronization stop operation if synchronization at said first synchronization point is not accomplished for said portion of processors within said expiration time, said synchronization stop operation marking an end of the synchronization period for said first processor; and

wherein each processor communicates with other processors by sending messages using a message switch.

22. (Original) The method of Claim 21, wherein each processor is an endpoint in said data storage system.

23. (Original) The method of Claim 22, wherein said synchronization start operation, synchronization check operation, and synchronization stop operation are operations used in defining a synchronization primitive used in the data storage system to coordinate processing between multiple processors in said data storage system.

24. (Original) A computer program product for synchronizing a plurality of endpoints in a data storage system at a first synchronization point, the computer program product comprising:

executable code that performs, by a first endpoint, a synchronization start operation wherein a first message is sent from said first endpoint to one or more other endpoints in the data storage system, said first message including a first key value corresponding to said first synchronization point representing a current processing state of said first endpoint;

executable code that determines, by said first endpoint, a timeout period;

executable code that determines, by said first endpoint using processing state information as reported to said first endpoint by other endpoints, whether synchronization with a selected portion of said one or more other endpoints at said first synchronization point has been accomplished within said timeout period; and

executable code that, if said first endpoint determines that synchronization has not been accomplished within said timeout period, sends a second message to said one or more other endpoints indicating that said first endpoint is at another synchronization point different from said first synchronization point.

25. (Currently Amended) The computer program product of Claim 24, wherein said plurality of endpoints are directors in said data storage system, said synchronization start operation defines a start of a synchronization period for the first endpoint, and said second message is sent as part of a synchronization stop operation marking an end of the synchronization period for the first endpoint.

26. (Original) The computer program product of Claim 25, wherein said first and second messages are synchronization messages sent using a message switch included in said data storage system.

27. (Original) The computer program product of Claim 26, further comprising:

executable code that stores, by said first endpoint, said processing state information in a key value storage area local to said first endpoint.

28. The computer program product of Claim 27, further comprising:

executable code that broadcasts, by each endpoint in the data storage system, a synchronization message including a key value corresponding to a current processing state of said each endpoint to every other endpoint in the data storage system; and

executable code that records, by each endpoint in the data storage system, key values for each endpoint for all received synchronization messages, each endpoint storing said key values in a key value storage area local to each endpoint.

29. (Original) The computer program product of Claim 28, wherein said executable code that determines whether synchronization with a selected portion of said one or more other endpoints at said first synchronization point has been accomplished within said timeout period is performed as part of a synchronization check operation.

30. (Original) The computer program product of Claim 29, wherein said second message is a synchronization message including a key value corresponding to said other synchronization point.

31. (Original) The computer program product of Claim 30, wherein said key value indicates one of: an invalid synchronization processing state, and another valid processing state of said first endpoint.

32. (Original) The computer program product of Claim 24, wherein said first endpoint determines whether synchronization has occurred for a subset of said selection portion of endpoints in accordance with said processing state information of said first endpoint.

33. (Original) The computer program product of Claim 24, wherein said selected portion of said one or more other endpoints is stored by said first endpoint as a synchronization mask.

34. (Original) The computer program product of Claim 33, wherein said synchronization mask includes a number of bits equal to a maximum number of possible endpoints in a data storage system.

35. (Original) The computer program product of Claim 24, wherein said timeout period is determined in accordance with an operation being performed for which said first synchronization point is specified.

36. (Original) The computer program product of Claim 24, wherein each endpoint in said data storage system maintains a local copy of state information representing a processing state of each endpoint as reported by each endpoint in synchronization messages sent over a message switch.

37. (Original) The computer program product of Claim 36, wherein each local copy of state information as maintained by each endpoint is not synchronized with other local copies of state information as maintained by other endpoints.

38. (Currently Amended) ~~The computer program product of Claim 37,~~

A computer program product for synchronizing a plurality of endpoints in a data storage system at a first synchronization point, the computer program product comprising:

executable code that performs, by a first endpoint, a synchronization start operation wherein a first message is sent from said first endpoint to one or more other endpoints in the data storage system, said first message including a first key value corresponding to said first synchronization point representing a current processing state of said first endpoint;

executable code that determines, by said first endpoint, a timeout period;

executable code that determines, by said first endpoint using processing state information as reported to said first endpoint by other endpoints, whether synchronization with a selected portion of said one or more other endpoints at said first synchronization point has been accomplished within said timeout period; and

executable code that, if said first endpoint determines that synchronization has not been accomplished within said timeout period, sends a second message to said one or more other endpoints indicating that said first endpoint is at another synchronization point different from

said first synchronization point, wherein each endpoint in said data storage system maintains a local copy of state information representing a processing state of each endpoint as reported by each endpoint in synchronization messages sent over a message switch, and each local copy of state information as maintained by each endpoint is not synchronized with other local copies of state information as maintained by other endpoints, and

wherein said first endpoint is any one of said plurality of endpoints which reaches said first synchronization point prior to other endpoints included in said selection portion in accordance with said processing state information, said processing state information being a private copy of synchronization information associated with said first endpoint.

39. (Original) The computer program product of Claim 38, wherein each of said endpoints in said selected portion perform synchronization in accordance with a peer to peer model.

40. (Original) The computer program product of Claim 24, wherein said first synchronization point represents an event corresponding to one of: enabling and disabling a mirrored write operation wherein when enabled, a write to a portion of a global memory is performed to two memory boards, and when disabled, a write to a portion of a global memory is performed to one memory board.

41. (Original) The computer program product of Claim 24, wherein said first synchronization point represents an event corresponding to one of: enabling and disabling operations to a hardware element.

42. (Original) The computer program product of Claim 41, further comprising:

disabling operations to said hardware element when said hardware element is brought

off-line.

43.   (Original) A computer program product for use in a data storage system for synchronizing execution at a first synchronization point between a plurality of processors in said data storage system, the computer program product comprising:

executable code that performs a synchronization start operation by a first processor defining a beginning of a synchronization period for said first processor, said first processor include executable code that performs said synchronization start operation including:

executable code that sends a first synchronization message to all other processors in said data storage system, said first synchronization message include a first key value corresponding to said first synchronization point representing a current processing state of said first processor;

executable code that determines a portion of said plurality of processors with whom said first processor is to synchronize at said first synchronization point;

executable code that determines whether synchronization at said first synchronization point has been accomplished for said portion of processors; and

executable code that determines an expiration period indicating a maximum amount of time for which said first processor will wait for synchronization at said first synchronization point for said portion of processors;

executable code that receives one or more other synchronization messages from other ones of said plurality of processors in said data storage system wherein each of said one or more other synchronization messages includes a key value indicating a processing state of its sending processor;

executable code that stores, by said first processor in a key value storage local to said first processor, key values from said one or more other synchronization messages, wherein said first

processor uses key values in said key value storage to determine whether synchronization at said first synchronization point has been accomplished for said portion of processors as part of a synchronization check operation; and

executable code that performs, by said first processor, a synchronization stop operation if synchronization at said first synchronization point is not accomplished for said portion of processors within said expiration time, said synchronization stop operation marking an end of the synchronization period for said first processor; and

wherein each processor communicates with other processors by sending messages using a message switch.

44. (Currently Amended) The computer program product of Claim [[44]] 43, wherein each processor is an endpoint in said data storage system.

45. (Currently Amended) The computer program product of Claim [[45]] 44, wherein said synchronization start operation, synchronization check operation, and synchronization stop operation are operations used in defining a synchronization primitive used in the data storage system to coordinate processing between multiple processors in said data storage system.